

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application of: C. EVANS

Attorney Docket No: 5231-096-US01

Application No.: 10/673,140

Group Art Unit: 2164

Filed: September 30, 2003

Examiner: R. Mahmood

For: SQL JOIN ELIMINATION

APPEAL BRIEF

Mail Stop APPEAL BRIEF-PATENTS

Commissioner for Patents

P.O. Box 1450

Alexandria, Virginia 22313-1450

Sir:

In response to the Notice of Panel Decision from Pre-Appeal Brief Review mailed December 29, 2009, Applicant submits the following Appeal Brief for entry and consideration by the Board of Patent Appeals and Interferences.

REAL PARTY IN INTEREST

The real party in interest in the present application is Oracle International Corporation, the assignee of the present application.

RELATED APPEALS AND INTERFERENCES

None.

STATUS OF THE CLAIMS

Claims 1-9 are currently pending, and rejected, in the present application. Accordingly, the rejection of claims 1-9 is currently being appealed.

STATUS OF AMENDMENTS

In response to the final Office Action mailed April 24, 2009, Applicant submitted a Pre-Appeal Brief Request for Review. The participants of the Pre-Appeal Brief conference decided that Applicant is required to submit an Appeal Brief in accordance with 37 CFR 41.37. Accordingly, none of the claims have been amended after the date of the Final Office Action mailed April 24, 2009.

SUMMARY OF THE CLAIMED SUBJECT MATTER

According to one aspect, the present invention comprises a computer implemented method of preventing execution of unnecessary joins between tables in a database. Written Description at Page 2, lines 25-28; Claim 1. The method includes the step of presenting a Structured Query Language (SQL) statement to the database, where the SQL statement refers directly to a set of tables in the database. *Id.* at Page 2, lines 29-33; Claim 1. Then, a list of tables is prepared that have a potential to be used to return the set of results but that are not directly referred to by the SQL statement. *Id.* at Page 2, lines 34-36; Claim 1. Tables that must be accessed in order to return the set of results from the list in accordance with a predetermined set of rules are then removed. *Id.* at Page 3, lines 1-4; Claim 1. The execution of joins involving any of the tables remaining in the list is prevented. *Id.* at Page 3, lines 5-6; Claim 1. Finally, a set of results from the database based on the SQL statement is returned. *Id.* at Page 9, lines 8-12; Claim 1.

GROUND OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1 stands rejected under 35 U.S.C. § 101 because the claimed method is purportedly not tied to a particular machine. Claims 1-9 also stand rejected under 35 U.S.C. § 102 as being anticipated by U.S. Patent No. 6,640,221 to Levine (“Levine”) Accordingly, Applicant respectfully requests review of both of these grounds of rejections on appeal.

ARGUMENT

As set forth above, Applicant respectfully requests review of the rejection of claim 1 under 35 U.S.C. § 101 and claims 1-9 under 35 U.S.C. § 102 as being anticipated by Levine. As discussed in greater detail below, claim 1 satisfies the statutory requirements under § 101 and Levine fails to teach each and every feature of the present invention recited by claims 1-9.

The Rejection of Claim 1 Under 35 U.S.C. § 101

The Examiner contends that claim 1 is improper because it is not tied to a particular machine. This assertion is maintained, despite the fact that claim 1 specifically recites a “computer implemented method.” Applicants’ Response to Office Action filed February 19, 2009 at Page 2 and 4. Although the Examiner states that this is merely a “nominal recitation,” viewing the claim in its entirety rebuts this argument. In other words, the elements of claim 1 recite a database, SQL statements, and joins, all of which are related to computers – which of course are machines.

Accordingly, the recitation of “computer implemented” viewed in the context of the claim, satisfies the requirements of § 101. As such, Applicants respectfully request withdrawal of the rejection of claim 1.

The Rejections Under 35 U.S.C. § 102

In order to anticipate a claim, a reference must teach each and every feature recited by the claims. MPEP § 2131. Levine fails in this regard for at least the following reasons.

Levine discloses that SQL statements that include more than one join operation have two types of result sets: (i) a final result set; and (ii) an intermediate result set. Applicants’ Response to Office Action filed December 27, 2007 at Pages 4-5 and Applicants’ Response to Office Action filed July 14, 2008 at Pages 5-7. The intermediate result set is the intermediate table that is generated from just two of the tables (or one table and another intermediate result set) being joined in one of the join operations. *Id.* The final result set is the final table that is generated from the tables being joined after all the join operations are executed. *Id.* Levine attempts to manipulate the intermediate result sets by providing a system and method for configuring, sequencing, and viewing joins in a SQL query. *Id.*

The process can be summarized as follows. First, Levine provides a join object for each table. *Id.* The join object includes a table relationships table that lists related tables. *Id.* A first

“raw” list of tables in an SQL query is then provided. *Id.* A first table from the first list is then moved into a second “ordered” list, and the table is deleted from the first list. *Id.* If the first table (which is now in the second list) has a related table in the table relationships list, the related table is also moved from the first list to the second list, and the related table is deleted from the first list. *Id.* at Pages 4-5. This process is repeated until there are no tables remaining in the list. *Id.* In sum, the process disclosed by Levine uses no discretion. *Id.* Rather, it removes tables from the first list until no tables remain. *Id.* In other words, Levine merely discloses reordering lists. *Id.*

An example is helpful to clarify this matter. Consider the “raw” table and “related table” below. Applicants’ Response to Office Action filed July 14, 2008 at Pages 5-7.

Raw Table
A
B
A1

Related Tables	
A	A1

In this example, tables A and A1 are related. Thus, according to the description of Levine above, the tables will be reordered. *Id.* First, table A will be moved to the “ordered” list and deleted from the “raw” list. *Id.* Then, the “Related Tables” list will be referenced to determine if any tables are related. *Id.* In this example, tables A and A1 are related. *Id.* Thus, table A1 will be moved from the “raw” table to the “ordered” table, and deleted from the “raw” table. *Id.* Because there are no more tables related to table A, table B will be moved to the “ordered” list. *Id.* Since table B does not have any related tables, the process described by Levine ends. *Id.* The final “ordered” table is shown below. *Id.*

Ordered Table
A
A1
B

The example shown above clarifies the operation discussed by Levine. *Id.* However, those skilled in the art will note that the related table serves merely as a reference table. *Id.* That is, it is not changed, *i.e.*, it still lists tables A and A1 as being related. *Id.* Indeed, Levine does not teach or suggest deleting any tables from the reference table. *Id.* Instead, tables are deleted from the “raw” table and moved to the “ordered” list. *Id.* Further, it is clear that the tables listed in the “Related Tables,” *i.e.*, table A and A1, are still joined in the “ordered” list. *Id.*

Independent claim 1 of the present application contrasts with Levine because it recites that one aspect of the present invention prevents “execution of joins involving any of the tables remaining in the list.” Applicants’ Response to Office Action filed February 19, 2009 at Page 5. As support for the rejection of the claims based on Levine, the Examiner makes blanket citations of large portions of Levine. *Id.* Additionally, the Examiner states that “since the tables required by the SQL statement are selected from the original list, the rest of the tables in the list do not participate in the SQL statement and are prevented from participating in the execution of joins.” *Id.* The Examiner’s contention, however, is explicitly rejected by the disclosure of Levine. Specifically, Levine states that the reordering of tables from the “raw” list to the “ordered” list (sets 162-172) is “repeated until all of the join objects have been moved from the ‘raw’ list to the ‘ordered’ list...” *Id.*

The direct, explicit, and unambiguous teaching of Levine directly contradicts the basis for the Examiner’s rejection of the claims of the present application. Thus, Levine fails to teach each and every feature of the present invention as recited by the claims. As such, Levine does not anticipate the present invention, and the rejection based thereon must be withdrawn. Accordingly, Applicants respectfully request issuance of a Notice of Allowance.

CLAIMS APPENDIX

1. (Previously Presented) A computer implemented method of preventing execution of unnecessary joins between tables in a database, the method comprising the steps of:
 - a. presenting a Structured Query Language (SQL) statement to the database, the SQL statement referring directly to a set of tables in the database;
 - b. preparing a list of tables that have a potential to be used to return the set of results but that are not directly referred to by the SQL statement;
 - c. removing tables that must be accessed in order to return the set of results from the list in accordance with a predetermined set of rules;
 - d. preventing execution of joins involving any of the tables remaining in the list; and
 - e. returning a set of results from the database based on the SQL statement.
2. (Previously Presented) The computer implemented method according to claim 1, wherein the predetermined set of rules includes allowing removal of a table from the list if the table is part of a join chain.
3. (Previously Presented) The computer implemented method according to claim 1, wherein the predetermined set of rules includes a rule allowing removal of a table from the list if the table forms the detail table in a join between a master table and a detail table.
4. (Previously Presented) The computer implemented method according to claim 1, wherein the predetermined set of rules includes a rule allowing removal of a table from the list if detail item values might not exist in a master table joined to a detail table.
5. (Previously Presented) The computer implemented method according to claim 1, wherein the predetermined set of rules includes a rule allowing removal of a table from the list if the table has a mandatory filter.

6. (Previously Presented) The computer implemented method according to claim 4, wherein the predetermined set of rules further includes a rule preventing removal of a table from the list that would otherwise be allowed, if the join is an outer join on a master table.
7. (Previously Presented) The computer implemented method according to claim 5, wherein the predetermined set of rules further includes a rule preventing removal of a table from the list that would otherwise be allowed, if the join is an outer join on a master table.
8. (Original) A computer program comprising computer program code means adapted to perform the steps of claim 1 when said program is run on a computer.
9. (Original) A computer program product comprising program code means stored on a computer readable medium for performing the method of claim 1 when said program product is run on a computer.

EVIDENCE APPENDIX

None.

RELATED PROCEEDINGS APPENDIX

None.

A Petition for Extension of Time is submitted herewith extending the time for response two months to and including March 29, 2010. No other fees are believed to be due at this time. Should any other fees be required, however, please charge such fee to Hanify & King, P.C., Deposit Account No. 50-4545, Order No. 5231-096-US01.

Respectfully submitted,
HANIFY & KING, P.C.

Dated: March 27, 2010

By: *Siddhesh V. Pandit*
Siddhesh V. Pandit, Registration No. 58,572
HANIFY & KING, P.C.
1055 Thomas Jefferson Street, N.W.
Suite 400
Washington, D.C. 20007
(202) 403-2104 Telephone
(202) 429-4380 Facsimile